A

Project Report on

## **Django To-do App**

Submitted in partial fulfillment of completion of the course

Advanced Diploma in IT, Networking and Cloud

Submitted by:

**Team Member-1: Manjima Bhattacharyya**

**Team Member-2: Anoushka Das**

Under Guidance of:

**Arpita Roy (Edunet)**

|  |  |  |
| --- | --- | --- |
|  |  |  |

Year 2022 -2024

1. Abstract
2. Acknowledgement
3. Team Composition and Workload Division
4. Table of Contents
5. Introduction to Problem
6. Proposed Solution
7. Requirements

7.1 Technology Stack

7.2Hardware

7.3 Software

7.4 Deployment Environment

8.User Requirements

9.Design Documentation

10.Implementation Details

11.Deployment

12.Conclusion

Appendix A Project Code

Appendix B Screenshot of Project

References

**ABSTRACT:**Daily planning has several benefits that can help you stay organized, focused, and productive.  Using a daily planner can help you with time management, improve your productivity, provide excellent stress relief, and offer a multitude of health benefits.  
  
There are a lot of reasons why having a daily planner is beneficial. The most important reason that you should have one is that it helps keep track of records of your daily activities. This includes homework, chores, appointments, and meetings. You can write all of your activities in your planner so that you'll never forget them. Having a daily planner is also essential for people who are always on the go. It's easy to keep track of things while you're running errands, going to a meeting, or attending a class. You can always pull out your planner and take a quick look when you're in a rush to make sure that you don't forget any important things.

Django is a high-level Python Web framework-based web framework that allows rapid development and clean, pragmatic design. Through this online daily planner, we can create notes like Google Keep or Evernote.

**ACKNOWLEDGEMENT**

Team ‘Trailblazers’ (comprising of 2 members, namely Manjima Bhattacharyya, Anoushka Das) are thankful to our teacher, Ms. Arpita Roy for her guidance and supervision which has provided a lot of resources needed in completing our project.

We are also thankful to the efforts put in by our team members and contributions to the preparation of this project.

**TEAM COMPOSITION AND WORKLOAD DIVISION:**

* **Manjima Bhattacharyya (Web Developer) (Leader)**: Responsible for designing and building the website, back-end development tasks, including coding, testing, and deployment. The team may consist of: - Back-end Developers: Responsible for building the server-side logic, managing databases, and implementing authentication and authorization.
* **Anoushka Das (Developer):** Contribution in building the website, including coding and deployment. Responsible for overseeing the project's planning, execution, and delivery, coordinate tasks, manage timelines, and ensure that the project stays on track and documentation and power point presentation.

**TABLE OF CONTENT**

**INTRODUCTION TO PROBLEM**

It is true that getting organized and using a daily planner won’t solve all of our problems, but it will make juggling them a little easier. Especially, when our screen time is high, having a digital or an on-screen planner makes it convenient and effective for individuals to use.

**PROPOSED SOLUTION**

Having a daily planner is also essential for people who are always on the go. It's easy to keep track of things while you're running errands, going to a meeting, or attending a class. You can always pull out your planner and take a quick look when you're in a rush to make sure that you don't forget any important things.

This will not only save time but also ensure increased productivity.

# REQUIREMENTS

# 1.Technology Stack:

# HTML, CSS, JAVASCRIPT

* DJANGO

# User authentication and authorization

# 2.Hardware Requirements:

# Processor: Intel(R) Core (TM) i5-9500 CPU @ 3.00GHz 3.00 GHz

# Installed RAM 8.00 GB (7.81 GB usable)

# System type 64-bit operating system, x64-based processor

# 3.Deployment Environment:

# Web server: Configure a web server to handle incoming request.

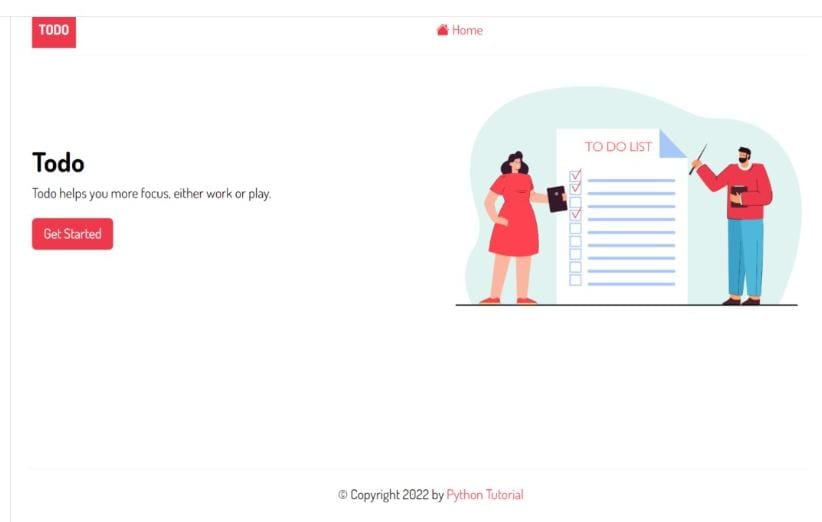
# Database Server: Set up a production ready database server with backup and radiancy.

**USER REQUIREMENT**

For all individual users, especially for students and working employees.

**DESIGN DOCUMENTATION**





**IMPLEMENTATION DETAILS**

1. Install the Django package - we need to install it using the following pip command: **pip install django**.

## Creating a new project-To create a new project todo\_list, you use the startproject command: **django-admin startproject todo\_list**

1. Creating a virtual environment **-** python command to [create a virtual environment](https://www.pythontutorial.net/python-basics/python-virtual-environments/) using the built-in venv module: **python -m venv venv**
2. And activate the venv virtual environment using the following command: **venv\scripts\activate.**
3. Creating a new project- To create a new project todo\_list, you use the startproject command**: django-admin startproject todo\_list**
4. Adding static files - create a static directory inside the project directory:
   1. **mkdir static**
5. Second, set the STATICFILES\_DIRS to the static directory in the settings.py after the STATIC\_URL file so that Django can find the static files in the static directory:
   1. **STATIC\_URL = 'static/'**
6. **STATICFILES\_DIRS = [BASE\_DIR / 'static']**
7. Third, create three directories js, css, and images inside the static directory:
   1. **cd static**
   2. **mkdir css images js**
   3. he static directory will look like this:
   4. ├── static
   5. | ├── css
   6. ├── images
   7. └── js
8. Setting up templates - create a templates directory inside the project directory: **mkdir templates**
9. Second, create a base.html template inside the templates.
10. Third, configure the template directory in the TEMPLATES of the settings.py file so that Django can find the base.html template.
11. Fourth, create the home.html template inside the template’s directory:
12. Creating a to-do application - First, create a to-do app in the todo\_list project using the startapp command: django-admin startapp to-do
13. Second, register the to-do app in the settings.py of the todo\_list project by adding it to the INSTALLED\_APPS list:
14. Third, create the templates directory inside the todo app directory:
15. **cd to-do mkdir templates**
16. Fourth, create the to-do directory inside the templates directory. The directory name must be the same as the app name.- **cd templates mkdir to-d**

DEPLOYMENT

The internet is a hostile environment. Before deploying your Django project, you should take some time to review your settings, with security, performance, and operations in mind.

Django includes many [security features](https://docs.djangoproject.com/en/5.0/topics/security/). Some are built-in and always enabled. Others are optional because they aren’t always appropriate, or because they’re inconvenient for development. For example, forcing HTTPS may not be suitable for all websites, and it’s impractical for local development.

Performance optimizations are another category of trade-offs with convenience. For instance, caching is useful in production, less so for local development. Error reporting needs are also widely different.

The following checklist includes settings that:

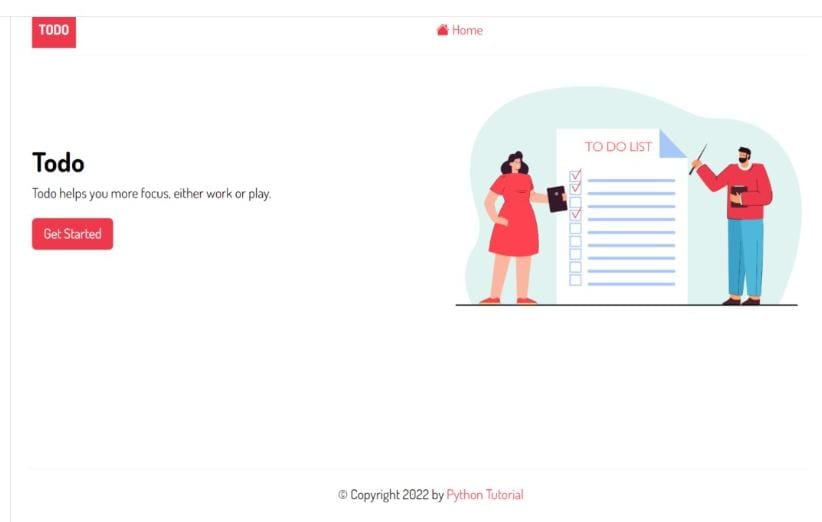
* must be set properly for Django to provide the expected level of security;
* are expected to be different in each environment;
* enable optional security features;
* enable performance optimizations;
* provide error reporting.

CONCLUSION

Django is an open source web application frame work which is written in Python. The Django framework gives us a simple and reliable way to create the course management system. It provides powerful functionalities and concise syntax to help programmers deal with the database, the web page and the inner logic.

This project report will help us develop insights on how to build part of a course management system using the Django framework. This project shows how to build a daily planner using the Django framework.

APPENDIX A: SCREENSHOT OF THE PROJECT





REFERENCES